Diyala Journal of Engineering Sciences

Journal homepage: https://djes.info/index.php/djes

ISSN: 1999-8716 (Print); 2616-6909 (Online)

Student Sentiment Analysis with Co-Curricular Activities and Placement Using Logistic Graph Convolution Neural Network

S. Johnbosco[1*], Ravi Lourdusamy[2], Denis R[3], Peter Jose P[4], P. Sathish[5], Mithun D'Souza[6]

[1,2]Department of Computer Science, Sacred Heart College (Autonomous), Affiliated to Thiruvalluvar University, India
[3,4]Department of Computer Science, Mount Carmel College, Autonomous, Bengaluru, Karnataka, India
[5]Department of Master of Computer Applications, Nitte Meenakshi Institute of Technology, India.
[6]Department of Computer Science, CHRIST (Deemed to be University), Bangalore, Karnataka, India

ARTICLE INFO

Article history:
Received 24/11/2025.
Revised: 01/05/2026,
Accepted: 13/06/2026,
Available online: 15/06/2026.

Keywords:
Deep Learning
Co-curricular activities
Euclidean Synthetic Minority Over - sampling
L1-regularised
Graph Convolution Neural Network

ABSTRACT

Student feedback sentiment analysis for co-curricular activities, employing deep learning, ascertains and classifies job placements based on students' feedback. Though several works have focused on student sentiment analysis, there remains room to analyze job placement and the effects of co-curricular activities on students' academic performance. This work proposes a method, called L1-Regularized Graph Convolutional Neural Network (L1-GCNN), to identify the relationship between co-curricular activities and students' placement performance. Initially, the raw student placement data is considered as input. After that, Euclidean Synthetic Minority Over-sampling-based pre-processing, including normalization and class balancing, is applied to the student placement dataset. Synthetic samples are generated efficiently, thereby avoiding distortions in local distributions and achieving class-balanced results. Following this, a normalized, class-balanced sample is used to train an L1-regularised Graph Convolutional Neural Network-based Student sentiment analysis, which is then applied to identify the most representative and optimal feature subset for estimating the impact of co-curricular activities on students' academic performance. Employing the graph structure, the L1-regularised or logistic function highlights the interdependencies between co-curricular metrics and student placement performance via feedback. Moreover, the L1-regularisation function improves the student performance system by optimizing the network and utilizing features more effectively. Experiments are conducted on the proposed L1-GCNN and existing methods using several metrics. The simulation results show relative improvements in student sentiment analysis for placement performance based on co-curricular activities, reducing the mean absolute error by 0.053, precision by 0.9, recall by 0.93, and accuracy by 0.89 compared to existing methods.

1. INTRODUCTION

There is sufficient evidence in the literature that co-curricular activities positively influence holistic development and the development of successful engineers. Both employers and students have attributed co-curricular activities to being a significant means of developing the proficiencies employers prefer in prospective engineering recruits. A notable body of educational research has expressed a positive relationship between student involvement in co-curricular activities and enhanced student outcomes regarding academic performance. Student sentiment analysis focused on co-curricular activities and placement uses sentiment analysis to understand students' emotions and opinions about their extracurricular experiences and career outcomes, helping educational institutions identify areas for improvement in student satisfaction, engagement, and career readiness.

Physical education (PE) has been a core element of holistic learning throughout history, directly linking physical activity to improved emotional regulation, mental health, and cognitive performance. Dai [1] proposed a hybrid algorithm combining K-means and the Improved Krill Swarm Algorithm (KSA) to improve students' learning potentialities and fine-tune teachers'

* Corresponding author's E-mail: boscosofts@gmail.com
DOI: 10.24237/djes.2026.19205

instructional strategies, ensuring an optimized process for improving recall and accuracy; however, evaluating students in physical education remained challenging. Liang and Liang [2] proposed an MAEM (Multi-Attribute Evaluation Model) based on deep learning to provide a comprehensive assessment method for physical education by combining data from wearable devices and self-reports, but this method failed to perform feature selection or address co-curricular activities. To address this, Hemdanou et al. [3] provided a detailed analysis of feature selection methods for student performance using distinct machine learning models, and Williams et al. [4] made a detailed evaluation of co-curricular activities to optimize student learning. Moreno Cortez and Nevarez Montes [5] investigated the connection between experiences in certain co-curricular activities and academic achievement, though several such studies fail to distinguish between activity types or account for time dedicated to each activity. Strielkowski et al. [6] conducted a bibliographic study on adaptive learning and artificial intelligence (AI), but without addressing data uncertainty and fuzziness; to focus on this, Li [7] designed a fuzzy decision support system that achieved high prediction accuracy and significantly reduced the error range. Olewnik et al. [8] investigated undergraduate students' involvement in co-curricular activities and their perceptions of its value and costs, while Pürgstaller et al. [9] emphasized aspects of digital media use in physical education. Weng et al. [10] employed deep learning to analyze academic performance and assess critical thinking and problem-solving potential, thereby enhancing teaching quality, and Yu and Qi [11] presented a hybrid mechanism combining bidirectional long short-term memory (BiLSTM) and an attention mechanism to integrate multimodal data. Nevertheless, prediction methods should also consider external factors alongside learning activities; to address this, Leelaluk et al. [12] introduced an Attention-Based Artificial Neural Network (Attn-ANN) to improve predictive performance, though the MAE was not reduced.

Rahman et al. [13] presented a machine learning-based analysis of the influences of co-curricular activities on academic performance, but failed to consider extracurricular activity courses. To focus on this, Ileritürk [14] designed a study examining co-curricular activities, both professional and non-professional, though accuracy did not improve.

The primary objective of the present study is to develop an L1-regularised Graph Convolutional Neural Network (L1-GCNN), supported by Euclidean Synthetic Minority Over-sampling (ESMOS)-based preprocessing, for analyzing student feedback sentiment on co-curricular activities and evaluating placement performance using the student placement dataset. The key contributions of this work are: (i) an ESMOS-based preprocessing and class-balancing pipeline that reduces the Mean Absolute Error (MAE) and produces normalized, class-balanced samples; (ii) an L1-regularised GCNN that performs feature selection and sentiment classification, identifying the most prominent co-curricular features and improving precision and recall; and (iii) experimental validation on two benchmark datasets, demonstrating improved accuracy, precision, recall, and MAE over existing K-means + Improved KSA [1], MAEM [2], and Attn-ANN [12] baselines.

The remainder of this paper is organized as follows: Section 2 describes recent work on student sentiment analysis using machine learning and deep learning. Section 3 provides a detailed description of the proposed method for validating placement performance using deep learning. Section 4 presents the experimental setup with detailed qualitative and quantitative analysis, including tables and graphical comparisons with existing methods. Finally, Section 5 concludes the paper.

2. LITERATURE REVIEW

Sentiment analysis based on aspects is paramount for extracting valuable frames of reference from textual data, particularly in educational contexts where understanding student feedback is pivotal to evaluating courses. The hybridization of co-curricular activities focused on English is paramount in higher education because it aims to boost students' interpersonal skills and language proficiency.

Le [15] presented the primary variables that deter active engagement among university students, identifying significant barriers including limited time availability and restricted English-language competency. Milicevica et al. [16] presented an in-depth examination of hackathon efficiency, with a focus on students' attitudes and teachers' involvement in educational methodology. Kalita et al. [17] predicted student academic performance using a Bi-LSTM deep learning framework with SHAP-based interpretability and statistical validation. Kashif Salee et al. [18] proposed a multi-task learning framework involving academic and co-curricular activities based on sentiment analysis for course evaluation, achieving the highest recall in aspect classification. Pan et al. [19] surveyed academic performance prediction using machine learning approaches.

There is strong evidence that co-curricular experiential learning activities positively influence the development of rounded, successful engineers. Mamani-Coaquira and Villanueva [20] conducted a holistic review of text sentiment analysis using ML and DL techniques. Hueck et al. [21] conducted a detailed review of co-curricular experiential learning integration and student participation across institutions.

Universal tools such as virtual classrooms, multimedia presentations, and learning management systems are now widely used in teaching and learning; nevertheless, AI techniques are not yet frequently utilized in higher education. Motivated by social constructivist theory, Singh et al. [22] surveyed the relationship between AI use and collaborative learning and its influence on overall performance in an online learning environment. Ertuğrul and Bitirim [23] conducted a systematic literature review of job recommender systems employing sentiment analysis of students' feedback. Williams et al. [4] conducted a detailed analysis of the impact of co-curricular learning on student choice.

Thelma et al. [24] conducted a hypothesis test analyzing students' participation in extracurricular activities and presented the results in detail. O'Donnell et al. [25] examined the relationship between extracurricular activity participation, school belonging, and depressed mood during adolescence, testing the compensation hypothesis. Lin et al. [26] analyzed the classification of sentiments for modelling student qualitative feedback, focusing on accuracy. Hooshangi et al. [27] designed an approach integrating practical computing skills and co-curricular activities into the curriculum, examining the impact on overall student experience. Wang and Wang [28] designed a Modality-Uncertainty-aware Knowledge Distillation Framework (MUKDF) to handle uncertainty and complexity in multimodal sentiment analysis; however, performance gains failed to materialize under both complete and incomplete modality conditions. Saidi et al. [29] introduced an Optimal Architecture for a Sentiment Analysis Transformer using Multihead Attention and Genetic Crossover, enhancing the accuracy and efficiency of sentiment analysis tasks. Yu and Yu [30] developed a mobile learning system based on convolutional network technology to improve the personalization and intelligence of learning resource recommendations.

Evaluating placement performance with co-curricular activities is a complicated procedure requiring scientific research and careful analysis because more emphasis must be placed on co-curricular activities rather than academics, with precise research recommending a path for increased sentiment analysis. In earlier research, researchers have endeavored to reveal associations between statistical analysis and co-curricular activities; however, for the most part, they have done so by using statistical methods.

Based on this review of related work, constrained traditional deep learning solutions have been designed to assess placement performance in co-curricular activities using student sentiment. Instead of focusing on the entire dataset, Euclidean Synthetic Minority Over-sampling-based preprocessing is applied to minimize the MAE, and an L1-Regularised Graph Convolutional Neural Network-based student sentiment analysis model is proposed. The L1-GCNN method is described in detail in the following sections.

3. METHODOLOGY

Based on extensive research utilizing sentiment analysis of student feedback, co-curricular activities harmonize with preferred placement outcomes. Positive student sentiment toward co-curricular activities is a yardstick of greater student engagement and the development of valuable skills that employers particularly evaluate during the placement process. The conventional GCNN method failed to reduce noise and prevent overfitting. Compared to existing GCNN-based methods, L1-regularised Graph Convolutional Neural Networks (L1-GCNNs) offer a superior approach to student sentiment analysis by handling noisy, sparse, and multi-relational academic data (feedback, co-curriculars, and placement results). Student datasets contain sparse connections (not every student participates in every activity or provides detailed feedback). L1 regularisation promotes sparsity by shrinking less important feature weights to zero, effectively reducing noise and preventing overfitting. L1 regularisation automatically selects the most relevant, non-redundant features to determine sentiment, outperforming traditional GCNs by eliminating irrelevant data. By combining textual feedback (placement results, co-curricular engagement), L1-GCNN aims to provide better sentiment categorization (positive, negative, and neutral) than conventional GCNNs. Figure 1 shows the block diagram of the L1-GCNN method.

Figure 1 shows that the L1-GCNN method is split into four parts. They are collecting data using a benchmark dataset, pre-processing, feature selection, and classification to evaluate student feedback sentiment analysis and analyze the effect of co-curricular activities on placement performance. First, Student Feedback Sentiment Analysis is applied to the Academic and Placement Performance dataset, and the raw data are fed as input to the Euclidean Synthetic Minority Over-sampling-based Pre-processing model. Normalization and class balancing are performed separately before the following process. Finally, an L1-Regularised Graph Convolutional Neural Network-based Student sentiment analysis is applied to obtain the most representative features for ascertaining the influence of co-curricular activities on placement performance. Finally, performance analysis is conducted to assess the method's efficiency.

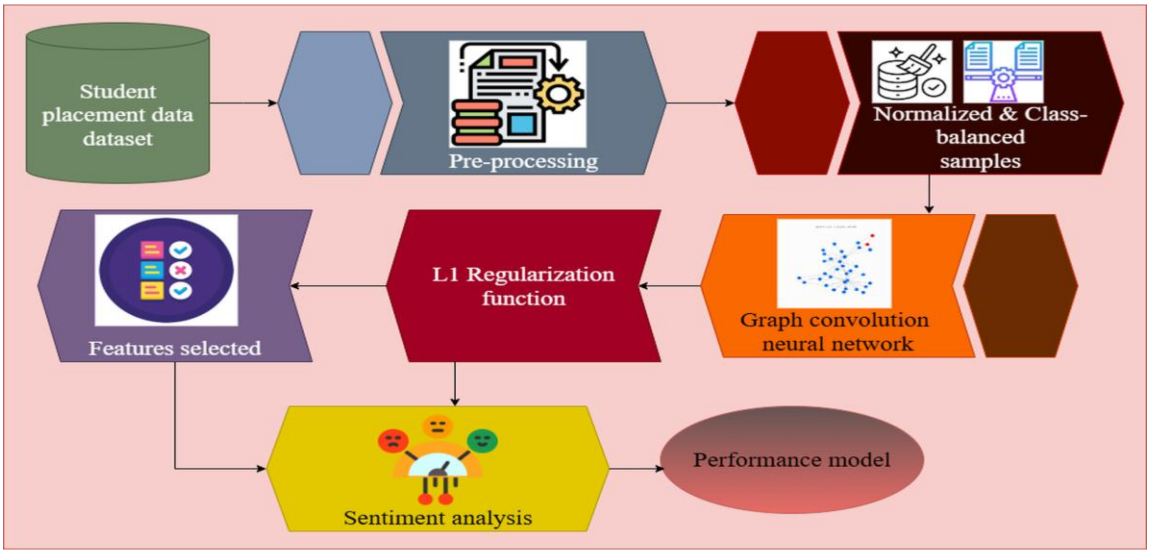

Figure 1. Block diagram of the Logistic Graph Convolution Neural Network (L1-GCNN) method

Dataset Description

The Student Feedback Sentiment Analysis with Academic and Placement Performance dataset, extracted from https://www.kaggle.com/datasets/koshikasaiprasad/student-placement-data/data [31], used in this study remains a popular tool for evaluating placement, mainly because it includes a broad set of parameters, such as academic and co-curricular activities. These are important for studying the function of students' feedback on sentiment analysis in building strong placement performance. This results in detailed performance metrics, such as academic esteem, employer esteem, faculty-student sentiment analysis, feedback ratio, and more, making it easier to scrutinize a student's sentiment. Also, the percentages secured in programming concepts, software engineering, and so on provide data on sustainability and ranking by percentage, in addition to co-curricular activities, which are crucial for assessing the role of education in generating innovation and soft power. The dataset enables academic and co-curricular comparisons, organized by attributes such as public speaking points, extra courses, hackathons, workshops, Olympiads, interesting games, and interesting types of books, among others. Publicly available on Kaggle, it supports educational institutions, researchers, and educators in benchmarking and strategic planning for academic excellence worldwide.

The Student Feedback Sentiment Analysis with Academic and Placement Performance dataset comprises 6 categories: teaching, course content, examinations, lab work, library facilities, and extracurricular activities. Each category's data consists of two columns, each with three labels: 0 (neutral), 1 (positive), or -1 (negative). The Student Feedback Sentiment Analysis with Academic and Placement Performance, called student placement data, is extracted from https://www.kaggle.com/datasets/koshikasaiprasad/student-placement-data/data [31]. The dataset includes 20,000 sample records and 39 features. Using this set of sample records and features, placements were made based on the suggested job role, which served as the final class labels. Table 1 below describes the dataset.

Another dataset, the College Student Placement Factors Dataset, is used in this study. The dataset is extracted from https://www.kaggle.com/datasets/sahilislam007/college-student-placement-factors-dataset [32]. The dataset consists of 10,000 college students and 10 .

Table 1. Student Feedback Sentiment Analysis dataset features

S. No	Features	S. No	Features	S. No	Features	S. No	Features
1	Academic percentage in Operating Systems	11	Logical quotient rating	21	Olympiads	31	Salary Range Expected
2	percentage in Algorithms	12	Hackathons	22	Reading and writing skills	32	In a relationship?
3	Percentage in Programming Concepts	13	Coding skills rating	23	Memory capability score	33	Gentle or Tough behaviour?

4	Percentage in Software Engineering	14	Public speaking points	24	Interested subjects	34	Management or Technical
5	Percentage in Computer Networks	15	Can work for a long time before the system?	25	Interested career area	35	Salary/work
6	Percentage in Electronics Subjects	16	Self-learning capability?	26	Job/Higher Studies?	36	Hard/smart worker
7	Percentage in Computer Architecture	17	Extra courses did	27	What type of company wants to settle in?	37	Have you ever worked in teams?
8	Percentage in Mathematics	18	certifications	28	Taken inputs from seniors or elders	38	Introvert
9	Percentage in Communication Skills	19	workshops	29	interested in games	39	Suggested Job Role
10	Hours worked per day	20	talenttests taken?	30	Interested in Types of Books		

It comprises features such as IQ, academic performance, CGPA, internships, communication skills, and more. The dataset is ideal for predictive modelling of placement outcomes, educational exercises in classification, feature importance analysis and End-to-end machine learning projects. Table 2 shows the attributes of the College Student Placement Factors Dataset**.**

Table 2. College Student Placement Factors Dataset Features

S. No	Features	Descriptions	S. No	Features	Descriptions
1	College_ ID	Unique ID of the college (e.g., CLG0001 to CLG0100)	**6**	Internship_ Experience	Whether the student has completed any internship (Yes/No)
2	IQ	Student's IQ score (normally distributed around 100)	**7**	Extra_ Curricular_ Score	Involvement in extracurricular activities (score from 0 to 10)
3	Prev_Sem_Result	GPA from the previous semester (range: 5.0 to 10.0)	**8**	Communication_ Skills	Soft skill rating (scale: 1 to 10)
4	CGPA	Cumulative Grade Point Average (range: ~5.0 to 10.0)	**9**	Projects_ Completed	Number of academic/technical projects completed (0 to 5)
5	Academic_ Performance	Annual academic rating (scale: 1 to 10)	**10**	Placement	Final placement result (Yes = Placed, No = Not Placed)

The dataset is represented as a structured matrix 'M', where each row corresponds to a unique sample and each column corresponds to a feature.

$$M = (S_1, S_2, \dots, S_m) \tag{1}$$

From the above equation (1), let 'M' denote the '$m * n$' matrix formulated as given below.

$$M = \begin{bmatrix} f_{11} & f_{12} & \cdots & f_{1m} \\ f_{21} & f_{22} & \cdots & M_{2m} \\ \cdots & \cdots & \cdots & \cdots \\ f_{N1} & f_{N2} & \cdots & f_{Nm} \end{bmatrix} \tag{2}$$

From the above equation (2), 'f_{ij}' represents the value of the '$j-th$' feature for the '$i-th$' sample instance. The dataset presented involves students' feedback analysis and tagging these features for placement performance. These features include logical quotient rating, hackathons, coding skills rating, public speaking, long-term work experience, self-learning, extra courses, certifications, workshops, talent tests, Olympiads, reading/writing skills, extracurricular activities, sports, and so on, to provide placement to students based on the feedback provided as sentiments. This structured data helps evaluate educational institutions' placement performance and perceived accreditation worldwide. Natural Language Processing (NLP) is employed to suggest job roles for student sentiment analysis to ascertain student feedback. The student's skills and feedback are mapped. The system identifies students' skills in academic and co-curricular activities and maps them with the requirements of different job roles to provide personalized recommendations. NLP processes text-based comments to classify student sentiment as positive, negative, or neutral, helping determine their true passions and engagement levels for courses, instructors, or projects. Learning methods are employed to map the analyzed skill profiles and sentiments against the requirements of different job roles. The system suggests specific, high-demand job roles and identifies skill gaps, providing actionable, personalized career development advice. Figure 2 shows the interrelationship between the key pillars of the suggested job role.

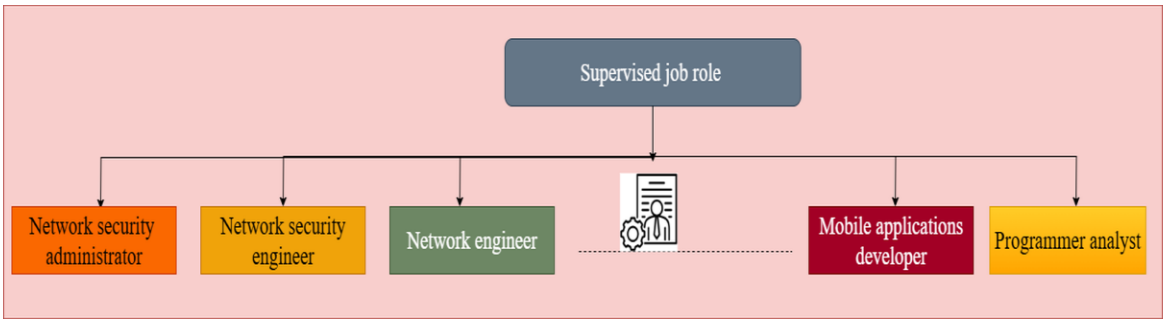

Figure 2. Holistic frameworks of the interconnected pillars of the suggested job role

By employing the above holistic, interconnected pillars of the suggested job role, as identified in the Student Feedback Sentiment Analysis with Academic and Placement Performance dataset, a detailed analysis is conducted to measure placement performance based on co-curricular activities.

Euclidean Synthetic Minority Over-sampling based Pre-processing.

Pre-processing is an analytical initial step in data analysis and deep learning that transforms raw data into a structured format appropriate for modelling. Real-world datasets frequently contain inconsistencies and irrelevant information that can obstruct model performance. Pre-processing techniques address these issues through normalization and balancing imbalanced datasets. Pre-processing improves predictive model accuracy by ensuring data quality and consistency, serving as the foundation for robust, meaningful analysis. No normalizations applied to standardize the features 'F' of the dataset 'DS'. For each feature 'j', the normalization function is formulated below.

$$f_{ij}^{norm} = \frac{f_{ij} - \min(f_j)}{\max(f_j) - \min(f_j)} \quad (3)$$

After performing the above normalization process in equation (3), the dataset becomes as given below.

$$M_{norm} = \begin{bmatrix} f_{11}^{norm} & f_{12}^{norm} & \cdots & f_{1m}^{norm} \\ f_{21}^{norm} & f_{22}^{norm} & \cdots & f_{2m}^{norm} \\ \cdots & \cdots & \cdots & \cdots \\ f_{N1}^{norm} & f_{N2}^{norm} & \cdots & f_{Nm}^{norm} \end{bmatrix} \quad (4)$$

Euclidean Synthetic Minority Over-sampling based Class Balancing

With the normalized samples as outputs, the distribution of classes (i.e., the supervised job role associated with placement performance) is significantly skewed, posing a substantial issue for deep learning classification tasks and leading to a biased model. To mitigate this issue, Euclidean Synthetic Minority Over-sampling-based Class Balancing is proposed in this work. Upon comparison to the conventional Synthetic Minority Over-sampling technique, which generates a new minority class via linear interpolation between the existing minority class. Though overfitting is addressed here, it restricts new samples to linear paths that may not accurately reflect the data distribution. To address this issue, the Euclidean Synthetic Minority Over-sampling employed in our work enhances the spatial scope for generating new classes by flexibly fine-tuning their stack pointers based on local density and distribution characteristics. Figure 3 shows the structure of Euclidean Synthetic Minority Over-sampling for class balancing.

As shown in Figure 2, the Euclidean distance between the random sample 'P_i' and its randomly selected adjacent sample 'P_j' is initially measured. The distance is multiplied using a random number ranging between '0' and '1' to obtain a random value. Following this, a ground sample 'P_{new1}' is obtained as given below.

$$P_{new1} = P_i + \alpha * (P_i - P_j) \quad (5)$$

A new sample is generated with the above results, as shown below.

$$P_{new} = \begin{cases} P_i + \gamma * (P_i - P_j) - \delta * (P_i - P_j), if\ Dis(P_{new1}, P_i) \geq 0.5 \\ P_i + \gamma * (P_i - P_j) + \delta * (P_i - P_j), if\ Dis(P_{new1}, P_i) < 0.5 \end{cases} \quad (6)$$

From the formula (6), according to the distance between 'P_{new1}' and the random sample 'P_i', a random value is added from the stack pointer of 'P_{new1}' to generate a new sample 'P_{new}'. If 'P_{new1}' is farther from 'P_{new}', the stack pointer of 'P_{new1}' is changed by obtaining the difference between a random value and generating 'P_{new}'. On the other hand, if 'P_{new1}' is closer to 'P_{new}', the stack pointer of 'P_{new1}' is changed by adding a random value to generate the 'P_{new}' result. The updated formula, in turn, ensures that the obtained samples are positioned around two selected samples. The Euclidean

Synthetic Minority Over-Sampling (ESMOS) balances class distributions by reducing noise and improving class balance, outperforming the conventional Synthetic Minority Over-Sampling Technique (SMOTE) in preserving spatial distributions and assigning advanced weights to clusters near the majority class, successfully generating synthetic samples for difficult-to-classify boundary cases. Compared to ADASYN, ESMOS prioritizes the spatial structure of minority clusters. While both aim to address SMOTE's limitations, ESMOS uses a different weighting mechanism based on the Euclidean distance to the majority class.

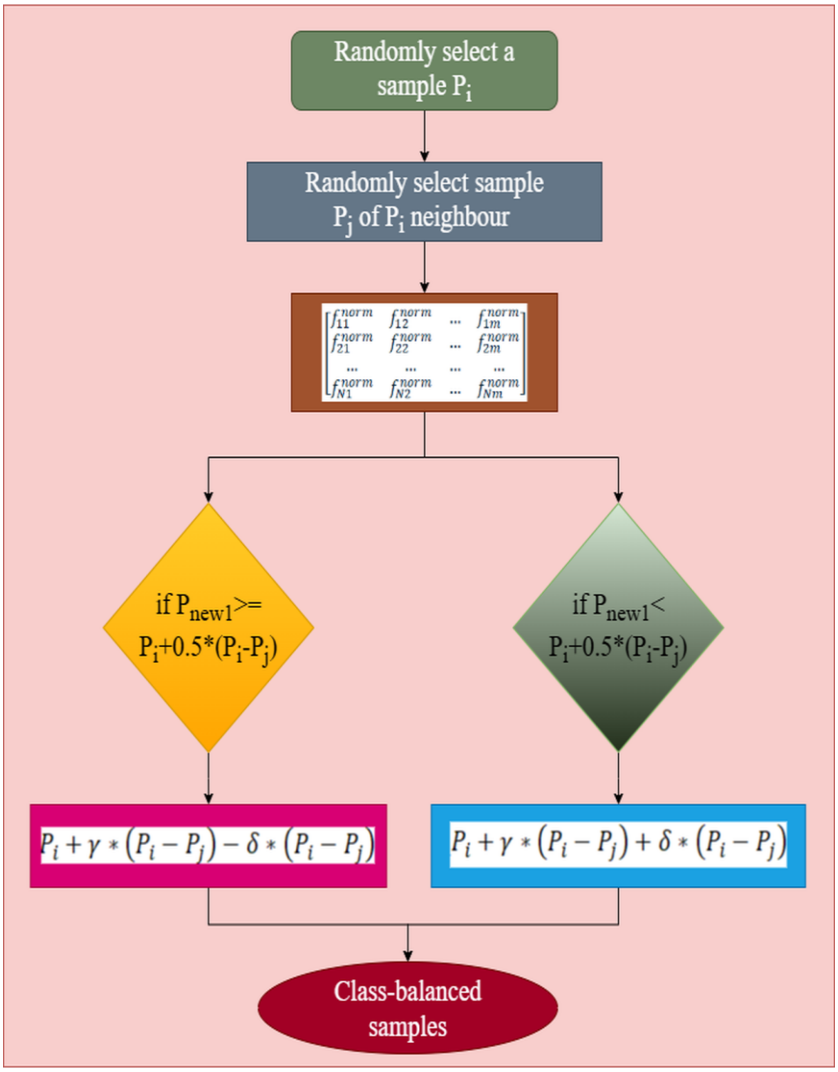

Figure 3. Structure of Euclidean Synthetic Minority Over-sampling based class balancing

L1-regularized Graph Convolution Neural Network-based Student sentiment analysis

Feature selection strives to retain features with high information content while eliminating those with low information content. In student sentiment analysis based on feedback to measure placement performance in co-curricular activities, specific features, such as percentages in software engineering, computer networks, or electronic subjects, may not adequately differentiate among variations within sample classes. Co-curricular activities contribute to the accuracy for predicting placement outcomes by demonstrating soft skills, such as leadership, teamwork, and communication, which employers often favour. While academic achievement predicts success, co-curricular engagement offers empirical evidence of practical abilities and holistic development, providing a more comprehensive profile of a candidate's potential and often yielding higher predictive accuracy in campus recruitment. To optimize model performance, feature selection uses a dependency measure between students and co-curricular activities to quantify the information each feature contributes to the overall sample. With this objective in mind, the L1-Regularized Graph Convolutional Neural Network model is applied to measure placement performance based on co-curricular activities. A Graph Neural Network (GNN) was preferred for sentiment analysis because it can explicitly model complex syntactic dependencies and contextual relationships between words, which sequential models like LSTMs or CNNs often miss, and because it provides a more structurally rich representation of text by capturing the underlying graph of semantic connections, leading to more accurate and explainable sentiment predictions. A graph structure for student-job role mapping is employed to enhance the classification performance by representing and learning from the intricate relationships between students and jobs. A Graph Convolutional Neural Network is used to measure the amount of information each feature contributes to the overall sample for accurate sentiment analysis. The normalized and class-balanced results are fed into the graph generation phase to ascertain the relationships among the data sources in student placement data obtained via feedback sentiment analysis. The graph consists of nodes, edges, and associations to capture the relationships among 39 features in the dataset.

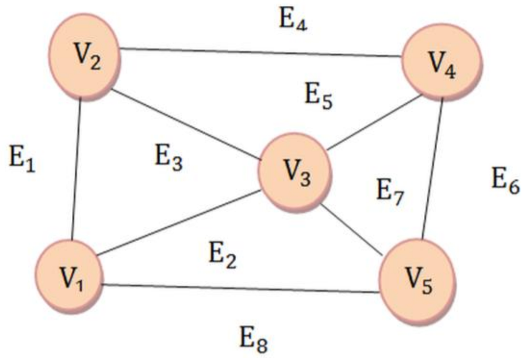

Figure 4. Graph construction process

Figure 4 illustrates the graph construction. In the graph theory, $G(V,\ \mathrm{E})$, where V denotes a vertex (i.e., academic node captures academic information) and E denotes an edge (i.e., co-curricular node captures extracurricular information). Edges here represent connections between nodes to maintain correlation among student, academic, and co-curricular information.

Figure 4 demonstrates the graph construction with five vertices (i.e., nodes) V_1, $\mathrm{V2}, \mathrm{V}_3, \mathrm{V}_4, \mathrm{V}_5$ and eight edges (i.e., links) $\mathrm{E1}, \mathrm{E}_2$, $\mathrm{E}_3, \mathrm{E}_4$, $\mathrm{E5}, \mathrm{E6}, \mathrm{E7}, \mathrm{E}_8$. The mapping is performed in the graph theory using the following mathematical formula,

$$MF : V_i \rightarrow V_j \tag{7}$$

From the formula (7), 'MF' represents the mapping function, 'V_i' and 'V_j' indicate a node in the network. This work aims to understand the relationship between the student's course and co-curricular information to arrive at the placement via a supervised job role. Following, which essential edges are designed based on the placement? This connection models course dependencies and co-curricular information to assist student placement via supervised job-role-based sentiment analysis. Finally, the relationship edges are generated based on similarities in student performance. These edges ascertain the supervised job role with the same co-curricular performance. Figure 5 shows the sample for five students and feedback details (i.e., the feature information present in the dataset).

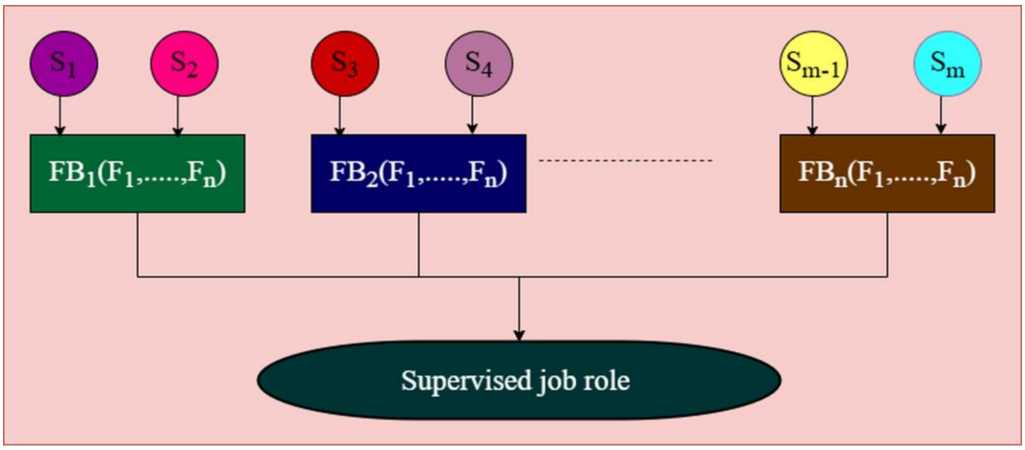

Figure 5. Sample graph for student placement via supervised job role based on sentiment analysis

Figure 5 shows that the proposed model consists of nodes and edges. These node and edge details are fed into the L1-Regularised or logistic-regularised Graph Convolutional Neural Network model to measure placement performance based on co-curricular activities. Initially, convolutional operations are applied to nodes to determine relationships among features. The graph data includes student placement data and student nodes, with relationships such as prerequisites and supervised job roles defined in edges. The supervised job roles are developing a system that matches students with jobs that align with their preferences (e.g., a desire for a positive work culture) and the sentiment expressed in job descriptions. This system improves the quality of job placements by matching students with roles that align with their expressed or inferred positive sentiment towards certain job aspects. Improve student satisfaction by providing more relevant and suitable job opportunities based on a deeper understanding of their preferences. The Graph Convolutional Neural Network (GCNN) model is designed for feature selection. It is graph-structured data. It includes multiple Graph Convolutional layers to aggregate multi-hop neighbour information and Fully Connected (FC) layers to perform classification. The input layer comprises two inputs: the adjacency matrix 'AM' and the feature matrix 'FM'. The dual input of the feature matrix and the adjacency matrix allows the model to simultaneously learn from both the content and the structure of the data. Multiple GCNN layers operate aggregation and transformation procedures. Fully connected layers create the placement performance. In the output layer, a sigmoid activation function is used to obtain the classified student sentiment feedback analysis results. The improvement in recall with the L1-GCNN method was achieved by applying a graph-based convolutional neural network. Applying this graph pattern helps determine the correlation between student courses and curricular information, from which the placement performance results were derived based on supervised job roles. This association between co-curricular information and placement performance, as revealed by sentiment analysis, yields results with minimal false negatives. The neural network comprises several graph convolutional layers for feature aggregation based on adjacent values. The neural network comprises two inputs, the adjacency matrix 'AM' and the feature matrix 'FM', and employs convolution operations for processing. The convolution procedure aggregates feature details from adjacent nodes to model a feature subset. The aggregation procedure is used in the feature selection process to obtain student features. The adjacent nodes are investigated, and each node's information is aggregated via the normalized adjacency matrix (NAM). The normalized adjacency matrix 'NAM' is mathematically formulated as given below.

$$NAM = DM^{-\frac{1}{2}}(AM + IM)DM^{-\frac{1}{2}} \quad (8)$$

$$DM = \sum_{ij}\left(AM_{ij} + IM_{ij}\right) \quad (9)$$

From Equations (8) and (9), the normalized adjacency matrix 'NAM' results from the diagonal matrix 'DM' and the identity matrix 'IM', respectively. The transformation process is applied to these aggregated results via a fine-tuned weight matrix 'WM' to generate more expressive node representations (i.e., relevant co-curricular-based features for placement in a supervised job role).

$$FM^{(l+1)} = Sigmoid\left(NAM.FM^{(l)}.WM^{(l)}\right) \quad (10)$$

From Equation (10), the convolution layer output result '$FM^{(l+1)}$' is obtained by activating via the sigmoid activation function '$Sigmoid$' based on the results of the normalized adjacency matrix 'NAM',

the feature matrix '$FM^{(l)}$', and the fine-tuned weight matrix 'WM', respectively. Unlike a single convolutional layer, multiple layers generate feature abstraction (i.e., relevant co-curricular-based features for placement in a supervised job role) by continuously updating layer outputs through aggregation and transformation procedures. The outputs generated are fed into fully connected layers to obtain the final outputs. The output layer is activated via an activation function to get the classified student sentiment feedback analysis results. Figure 6 shows the block diagram of the L1-regularised graph convolution neural network model.

As shown Figure 6, the input features 'F' are fed into the first layer, and the output 'Out_1' is fed into the next layer 'Out_2'. The generated output 'Out_2' is then fed into the fully connected layer 'FC', which is processed using the weight matrix 'WM' and bias 'b'. The output 'Out_3' is fed into the final output layer, which generates placement performance estimates based on sentiment analysis of student feedback. The mathematical formula is given below.

$$Out_1 = Sigmoid(WM_1 * DM^{-0.5} * (AM + IM) * DM^{-0.5} * F) \quad (11)$$

$$Out_2 = Sigmoid(WM_2 * DM^{-0.5} * (AM + IM) * DM^{-0.5} * Out_1) \quad (12)$$

From the above equations (11) and (12), the output results 'Out_1' and 'Out_2' are arrived at using the graphic convolution fine-tuning weight matrix 'WM_1' and 'WM_2' activated via the '$Sigmoid$' function. Also, the diagonal matrix 'DM', adjacency matrix 'AM', and identity matrix 'IM' are employed in obtaining the most representative features concerning input features 'F' via 'Out_1'and 'Out_2'. In our work, we use L1-regularisation for feature screening. This is achieved by setting irrelevant feature coefficients to zero and retaining the model's most valuable features. The L1-regularisation function is applied below to select the most representative features relating to co-curricular activities.

$$b_3 = \min_{\beta}\left\{\sum_{i=1}^{N}\left(Res_i - \sum_{j=1}^{M}\beta_j A_{ij}\right) + \lambda\left(\sum_{j=1}^{M}|\beta_j|\right)^2\right\} \quad (13)$$

From Equation (13), 'β_j' denotes the feature coefficient for the corresponding sample input 'A_{ij}' referring to the '$i-th$' student with respect to the '$j-th$' feature, and results of classified supervised job role stored in 'Res_i' using regularisation parameter 'λ' respectively.

$$Out_3 = Sigmoid(FC * Out_2 + b_3) \quad (14)$$

Following this, the student placement performance results 'Out_3' are generated via the fully connected layer based on the most relevant co-curricular features. The L1-regularised GCNN offers advantages, including automatic feature selection by pushing the minimum important node or edge weights to zero, resulting in sparser, simpler models. This sparsity improves model interpretability, as fewer features are considered, and improves robustness by decreasing the complexity of the model and sensitivity to noise. The automatic elimination of irrelevant features also helps achieve better generalization on unseen data and prevent overfitting. The pseudo-code representation of the L1-regularised Graph Convolutional Neural Network for sentiment analysis of student feedback on co-curricular activities and students' placement performance results is given below.

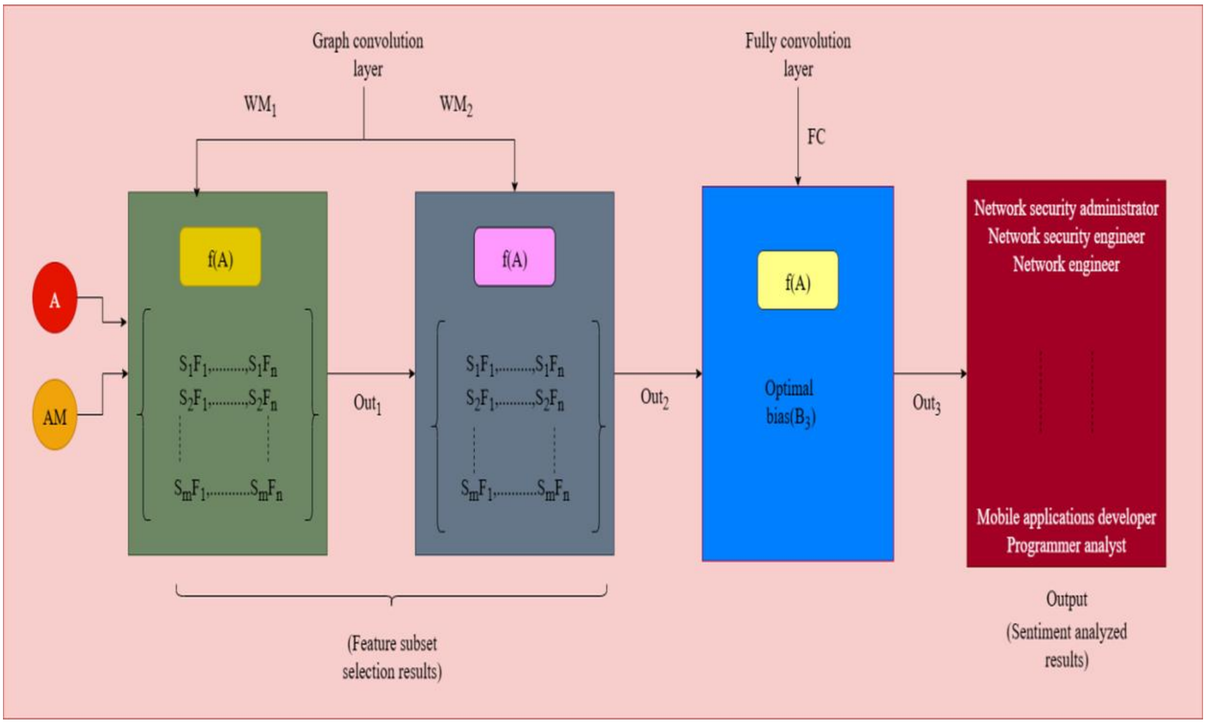

Figure 6. Block diagram of the L1-regularised graph convolution neural network model

Algorithm L1-regularised Graph Convolution Neural Network for analyzing student feedback on placement performance

Input: Dataset 'DS', Samples '$S = \{S_1, S_2, \dots, S_m\}$', Features '$F = \{F_1, F_2, \dots, F_n\}$', Classes '$C = \{C_1, C_2, \dots, C_u\}$'

Output: Robust sentiment analysis

1: **Initialization** '$m = 20000$', '$n = 39$', '$u = 34$', '$\gamma, \delta \in [0,1]$'
2: **Begin**
//**Normalized and Class-balanced Pre-processing**
3: **For**each Dataset 'DS' with Samples 'S', Features '$F = \{F_1, F_2, \dots, F_n\}$', Classes '$C = \{C_1, C_2, \dots, C_u\}$'
4: Formulate a structure matrix according to (1) and (2)
5: Generate normalization results according to (3) and (4)
6: Generate ground sample 'P_{new1}' according to (5)
7: Generate class-balanced new sample according to (6)
8: **Return** normalized and class-balanced results
9: **End for**
10: **For** each Dataset 'DS' with Classes '$C = \{C_1, C_2, \dots, C_u\}$', normalized and class-balanced results 'P_{new}'
11: Perform Mapping Function according to (7)
12: Evaluate results of normalized adjacency matrix according to (8) and (9)
13: Generate the convolution layer output result according to (10)
14: Evaluate results of 'Out_1' and 'Out_2' to generate the most representative features according to (11) and (12)
15: Generate L1-regularised optimal feature-selected output according to (13)
16: **Return** features selected results '$Out_2 \in FS$'
17: Generate classified results via a fully connected layer according to (14)
18: **Return** placement results (i.e., supervised job role)
19: **End for**
20: **End**

As given in Algorithm 1, the overall process for analyzing student feedback on placement performance is split into four sections. First, the raw student placement data is obtained and provided as input for further processing. Next, the raw student placement data is subjected to Euclidean Synthetic Minority Over-sampling-based Pre-processing. Here, normalization is performed first, followed by class balancing to standardize the features. In contrast to the traditional Synthetic Minority Over-sampling technique, which generates new minority class samples but confines them to linear paths, thereby not reflecting the true data distribution.

On the other hand, by using Euclidean Synthetic Minority Over-sampling, we can boost the spatial scope by flexibly fine-tuning their stack pointers based on local density and distribution characteristics, improving mean absolute error during sentiment analysis. Using normalized, class-balanced samples, the most representative and optimal feature subset is generated with an L1-regularised Graph Convolutional Neural Network. Here, aggregating and applying the transformation process along with the fine-tuned weights selects the most representative feature related to co-curricular activities. Finally, the most representative feature and class-balanced samples are used as input to the fully connected layer, which, with conditional checks, generates output results that accurately and precisely assign a supervised job role based on co-curricular activities.

Experimental setup

The proposed L1-regularised Graph Convolution Neural Network (L1-GCNN) method for analyzing student feedback and measuring placement performance according to co-curricular activities is explored and tested with other significant methods, namely a hybrid algorithm combining K-means and the Improved Krill Swarm Algorithm (KSA) (K-means and Improved KSA [1] and Multi-Attribute Evaluation Model (MAEM) [2] and Attention-Based Artificial Neural Network (Attn-ANN) [12]. An inducing feature for the evaluation metric is their potential to differentiate between results from different deep learning methods developed in Python, using the Student Feedback Sentiment Analysis with Academic and Placement Performance dataset and the College Student Placement Factors Dataset. Student Feedback Sentiment Analysis with Academic and Placement Performance dataset extracted from https://www.kaggle.com/datasets/koshikasaiprasad/student-placement-data/data [31]. The College Student Placement Factors Dataset is taken from https://www.kaggle.com/datasets/sahilislam007/college-student-placement-factors-dataset [32]. To conduct the experiments, the dataset is split into training and test sets. 70% of the samples are used for training, and the remaining 30% are used for testing. The efficiency of the deep learning method is evaluated using multiple execution metrics. For the proposed work, the method is validated using Mean Absolute Error (MAE), Precision, Recall, and Accuracy.

The objective of the proposed method is to achieve accurate sentiment analysis to evaluate placement performance using student feedback on co-curricular activities. Based on the objectives, the existing methods, such as K-means [1], Improved KSA [1], MAEM [2], and Attn-ANN [12], are taken as the base papers. These two base papers are explained to understand the proposed method. The proposed method concept is derived from the problems identified in these base papers. The drawbacks of these methods are effectively addressed by implementing the proposed method.

4. RESULTS AND DISCUSSION

In this study, we developed an L1-regularised Graph Convolutional Neural Network (L1-GCNN) to determine the relationship between co-curricular activities and students' placement performance. We compared our proposed L1-GCNN with existing K-means, Improved KSA [1], MAEM [2], and Attn-ANN [12] using two datasets and several metrics to validate the results.

Initially, Student Feedback Sentiment Analysis is performed using the Academic and Placement Performance dataset and the College Student Placement Factors Dataset, with their raw data as input. Second, Pre-processing is performed using Euclidean Synthetic Minority Over-sampling to obtain computationally efficient, class-balanced sample outcomes.

After that, an L1-Regularised Graph Convolutional Neural Network-based Student sentiment analysis is employed to select the most essential features and achieve the co-curricular activities performance.

Case study and Inferences

In this section, the case analysis of placement performance, employing students' sentiments regarding co-curricular activities, is simulated using the L1-GCNN method on the student placement dataset. First, Euclidean Synthetic Minority Over-sampling-based Pre-processing is applied to the sample instances. Figure 7 shows the application of sample instances from the student placement dataset to the suggested job role distributions.

As shown in Figure 7, the samples were normalized with supervised job role as input features. The Euclidean Synthetic Minority Over-sampling function was applied to generate Class-balanced results. Integrating the Euclidean function with the class-balancing function helps minimize MAE and reduce overfitting. According to the class-balanced results for the supervised job role, the most prominent features for analyzing sentiments are generated using an L1-Regularised Graph Convolutional Neural Network-based Student sentiment analysis. Here, the most prominent features and the placement performance validation results are selected. Figure 8 shows the results of the most prominent features selected based on high or low information.

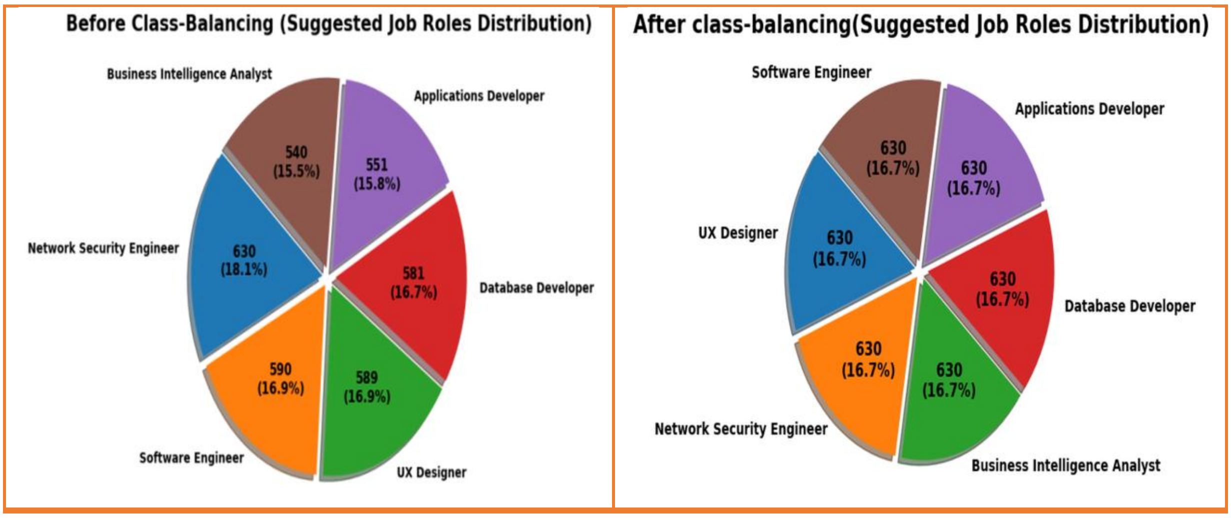

Figure 7. Euclidean Synthetic Minority Over-sampling Class-balanced results of supervised job role

	Feature	Weight	Category
0	memory capability score	0.011290	High Information (Selected)
1	Percentage in Programming Concepts	0.009763	High Information (Selected)
2	coding skills rating	0.007520	High Information (Selected)
3	olympiads	0.005663	High Information (Selected)
4	Interested Type of Books	0.005584	High Information (Selected)
5	interested career area	0.005198	High Information (Selected)
6	Type of company want to settle in?	0.004716	High Information (Selected)
7	percentage in Algorithms	0.003859	High Information (Selected)
8	workshops	0.003551	High Information (Selected)
9	public speaking points	0.002809	High Information (Selected)
10	interested in games	0.002114	High Information (Selected)
11	Logical quotient rating	0.001643	High Information (Selected)
12	hackathons	0.000100	High Information (Selected)
13	Percentage in Computer Networks	0.000100	High Information (Selected)
14	hard/smart worker	0.000100	High Information (Selected)
15	reading and writing skills	0.000100	High Information (Selected)
16	Percentage in Mathematics	0.000100	High Information (Selected)
17	Acedamic percentage in Operating Systems	0.000100	High Information (Selected)
18	Extra-courses did	0.000100	High Information (Selected)
19	self-learning capability?	0.000100	High Information (Selected)
20	can work long time before system?	0.000100	High Information (Selected)
21	In a Realtionship?	0.000000	Low Information
22	Salary/work	0.000000	Low Information
23	Salary Range Expected	0.000000	Low Information
24	Gentle or Tuff behaviour?	0.000000	Low Information
25	worked in teams ever?	0.000000	Low Information
26	Management or Technical	0.000000	Low Information
27	talenttests taken?	0.000000	Low Information
28	Taken inputs from seniors or elders	0.000000	Low Information
29	Job/Higher Studies?	0.000000	Low Information
30	Interested subjects	0.000000	Low Information
31	certifications	0.000000	Low Information
32	Hours working per day	0.000000	Low Information
33	Percentage in Communication skills	0.000000	Low Information
34	Percentage in Computer Architecture	0.000000	Low Information
35	Percentage in Electronics Subjects	0.000000	Low Information
36	Percentage in Software Engineering	0.000000	Low Information
37	Introvert	0.000000	Low Information

🎯 Selected High-Information Features :

	Feature
0	memory capability score
1	Percentage in Programming Concepts
2	coding skills rating
3	olympiads
4	Interested Type of Books
5	interested career area
6	Type of company want to settle in?
7	percentage in Algorithms
8	workshops
9	public speaking points
10	interested in games
11	Logical quotient rating
12	Acedamic percentage in Operating Systems
13	reading and writing skills
14	self-learning capability?
15	Extra-courses did
16	can work long time before system?
17	hackathons
18	Percentage in Mathematics
19	Percentage in Computer Networks
20	hard/smart worker

Figure 8. Prominent features selected results.

As shown in Figure 8, the twenty-one most prominent features are selected for further processing based on the differentiation between high- and low-information feature subsets. Employing the above prominent features using a graph convolutional neural network improves overall precision and recall. Finally, with class-balanced samples and using prominent features, sentiment analysis results for placement performance regarding co-curricular activities are provided in Figure 9. The classification conditions are based on prominent features and class-balanced samples.

```
Conditions for Suggested Job Role prediction:
|--- Logical quotient rating <= 0.06
|   |--- Percentage in Mathematics <= 0.30
|   |   |--- Percen tage in Programming Concepts <= 0.17
|   |   |   |--- memory capability score in ['excellent', 'medium']
|   |   |   |   |--- interested career area  in ['Business process analyst', 'cloud computing', 'developer']
|   |   |   |   |   |--- class: Applications Developer
|   |   |   |   |--- interested career area  in ['security', 'system developer', 'testing']
|   |   |   |   |   |--- class: UX Designer
|   |   |   |--- memory capability score in ['poor']
|   |   |   |   |--- percentage in Algorithms <= 0.22
|   |   |   |   |   |--- class: UX Designer
|   |   |   |   |--- percentage in Algorithms > 0.22
|   |   |   |   |   |--- class: Business Intelligence Analyst
|   |   |--- Percentage in Programming Concepts > 0.17
|   |   |   |--- interested career area  in ['Business process analyst', 'cloud computing']
|   |   |   |   |--- reading and writing skills in ['excellent', 'medium']
|   |   |   |   |   |--- class: Software Engineer
|   |   |   |   |--- reading and writing skills in ['poor']
|   |   |   |   |   |--- class: Network Security Engineer
|   |   |   |--- interested career area  in ['developer', 'security', 'system developer', 'testing']
|   |   |   |   |--- coding skills rating <= 0.31
|   |   |   |   |   |--- class: Database Developer
|   |   |   |   |--- coding skills rating > 0.31
|   |   |   |   |   |--- class: Business Intelligence Analyst
|   |--- Percentage in Mathematics > 0.30
|   |   |--- Percentage in Programming Concepts <= 0.87
|   |   |   |--- percentage in Algorithms <= 0.01
|   |   |   |   |--- class: Database Developer
|   |   |   |--- percentage in Algorithms > 0.01
|   |   |   |   |--- Interested Type of Books in ['Action and Adventure', 'Anthology', 'Art', 'Autobiographies', 'Biographies', 
'Childrens', 'Comics', 'Cookbooks', 'Diaries', 'Dictionaries', 'Drama', 'Encyclopedias', 'Fantasy', 'Guide', 'Health', 'History', 
'Horror', 'Journals', 'Math', 'Mystery', 'Poetry', 'Prayer books', 'Religion-Spirituality', 'Romance', 'Satire', 'Science']
|   |   |   |   |   |--- class: Applications Developer
|   |   |   |   |--- Interested Type of Books in ['Science fiction', 'Self help', 'Series', 'Travel', 'Trilogy']
|   |   |   |   |   |--- class: Network Security Engineer
|   |   |--- Percentage in Programming Concepts > 0.87
|   |   |   |--- Type of company want to settle in? in ['BPA', 'Cloud Services', 'Finance', 'Product-based', 'SAaS services', 'Sales 
and Marketing', 'Service-based']
|   |   |   |   |--- Acedamic percentage in Operating Systems <= 0.38
|   |   |   |   |   |--- class: Business Intelligence Analyst
|   |   |   |   |--- Acedamic percentage in Operating Systems > 0.38
|   |   |   |   |   |--- class: Applications Developer
|   |   |   |--- Type of company want to settle in? in ['Testing and Maintenance Services', 'Web Services', 'product 
development']
|   |   |   |   |--- hard/smart worker in ['hard worker']
|   |   |   |   |   |--- class: Business Intelligence Analyst
|   |   |   |   |--- hard/smart worker in ['smart worker']
|   |   |   |   |   |--- class: UX Designer
|--- Logical quotient rating > 0.06
|   |--- public speaking points <= 0.06
|   |   |--- Percentage in Computer Networks <= 0.63
|   |   |   |--- Acedamic percentage in Operating Systems <= 0.39
|   |   |   |   |--- Percentage in Mathematics <= 0.96
|   |   |   |   |   |--- class: UX Designer
```

```
| | | | |--- Percentage in Mathematics > 0.96
| | | | | |--- class: Applications Developer
| | | |--- Acedamic percentage in Operating Systems > 0.39
| | | | |--- workshops in ['cloud computing', 'data science', 'database security', 'game development']
| | | | | |--- class: Software Engineer
| | | | |--- workshops in ['hacking', 'system designing', 'testing', 'web technologies']
| | | | | |--- class: Applications Developer
| | |--- Percentage in Computer Networks > 0.63
| | | |--- coding skills rating <= 0.18
| | | | |--- Percentage in Programming Concepts <= 0.54
| | | | | |--- class: Business Intelligence Analyst
| | | | |--- Percentage in Programming Concepts > 0.54
| | | | | |--- class: Applications Developer
| | | |--- coding skills rating > 0.18
| | | | |--- Percentage in Mathematics <= 0.78
| | | | | |--- class: Database Developer
| | | | |--- Percentage in Mathematics > 0.78
| | | | | |--- class: Network Security Engineer
| |--- public speaking points > 0.06
| | |--- Percentage in Computer Networks <= 0.22
| | | |--- Interested Type of Books in ['Action and Adventure', 'Anthology', 'Art', 'Autobiographies', 'Biographies',
'Childrens', 'Comics', 'Cookbooks', 'Diaries', 'Dictionaries', 'Drama', 'Encyclopedias', 'Fantasy', 'Guide', 'Health', 'History',
'Horror']
| | | | |--- percentage in Algorithms <= 0.45
| | | | | |--- class: Applications Developer
| | | | |--- percentage in Algorithms > 0.45
| | | | | |--- class: Business Intelligence Analyst
| | | |--- Interested Type of Books in ['Journals', 'Math', 'Mystery', 'Poetry', 'Prayer books', 'Religion-Spirituality',
'Romance', 'Satire', 'Science', 'Science fiction', 'Self help', 'Series', 'Travel', 'Trilogy']
| | | | |--- self-learning capability? in ['no']
| | | | | |--- class: Applications Developer
| | | | |--- self-learning capability? in ['yes']
| | | | | |--- class: UX Designer
| | |--- Percentage in Computer Networks > 0.22
| | | |--- Percentage in Programming Concepts <= 0.04
| | | | |--- Percentage in Computer Networks <= 0.27
| | | | | |--- class: Network Security Engineer
| | | | |--- Percentage in Computer Networks > 0.27
| | | | | |--- class: Applications Developer
| | | |--- Percentage in Programming Concepts > 0.04
| | | | |--- workshops in ['cloud computing']
| | | | | |--- class: Database Developer
| | | | |--- workshops in ['data science', 'database security', 'game development', 'hacking', 'system designing', 'testing',
'web technologies']
| | | | | |--- class: Business Intelligence Analyst
```

Figure 9. Conditions for Suggested Job Role prediction of classification

Classification Results

Based on the conditions shown in the above figure, the classification results for the supervised job role are shown in Figure 10.

As shown in Figure 10, the suggested job role classifications are based on co-curricular activities. By emphasizing students' feedback on co-curricular activities, accuracy improved. The detailed quantitative analysis is provided in the following sub-section.

Discussion

Performance analysis of Mean Absolute Error

Mean Absolute Error (MAE) with sentiment analysis quantifies the average absolute difference between predicted and actual sentiment values for co-curricular activities, thereby measuring placement performance. This is mathematically formulated as given below.

$$MAE = \frac{1}{N}\sum_{i=1}^{N}(Pred_i - Act_i) \quad (15)$$

From equation (15), the Mean Absolute Error (MAE) is calculated based on the predicted sentiments ($Pred_i$) and the actual sentiments (Act_i) used to measure placement performance. Lower MAE indicates better sentiment analysis and vice versa. Table 3, given below, lists the MAE results for four methods: L1-GCNN, K-means, Improved KSA [1], MAEM [2], and Attn-ANN [12].

	Percentage in Programming Concepts	coding skills	Percentage in Computer Networks	Academic percentage in Operating Systems	percentage in Algorithms
0	0.15	0.75	0.50	0.97	0.35
1	0.32	1.00	0.71	0.12	0.50
2	0.06	0.12	0.94	0.68	0.12
3	0.03	0.75	0.88	0.53	0.76
4	0.44	0.50	0.59	0.47	0.03
...	...	...	...	...	...
3775	0.47	0.62	0.68	0.47	0.50
3776	0.35	0.25	0.47	0.26	0.74
3777	0.56	0.75	0.74	0.12	0.24
3778	0.65	0.38	0.44	0.97	0.94
3779	0.09	0.12	0.97	0.12	0.76

Percentage in Mathematics		Interested Type of Books	interested in games	Type of company want to settle in?		
0.42	Minimum percent analyst	Autobiographies	no	hackathons	...	0.33
0.19	testing	Health	yes	Sales and Marketing	...	0.83
0.45	Minimum percent analyst	Health	no	BPA	...	0.00
0.24	system developer	Self help	yes	BPA	...	0.83
0.46	developer	Mystery	no	Finance	...	0.17
...	...	...	...	Testing and Maintainance Services	...	...
0.18	system developer	Drama	yes	BPA	...	0.17
0.55	business process analyst	Guide	yes	Cloud Services	...	0.33
0.41	security	Journals	yes	Testing and Maintainance Services	...	0.17
0.38	cloud computing	Series	yes	SaaS services	...	0.00
0.30	testing	Drama	yes	Product based	...	0.00

memory capability score	reading and writing skills	can work long time before system?	self-learning capability?
medium	excellent	no	yes
medium	poor	yes	no
excellent	excellent	no	yes
poor	medium	yes	yes
poor	poor	no	yes
...	...	...	...
excellent	excellent	yes	no
medium	excellent	yes	yes
excellent	medium	yes	yes
poor	medium	no	yes
medium	medium	no	yes

Extra-courses did	olympiads	hard/smart worker	workshops	Suggested Job Role
no	yes	hard worker	cloud computing	Network Security Engineer
no	no	smart worker	system designing	Business Intelligence Analyst
yes	yes	hard worker	game development	Applications Developer
no	no	smart worker	game development	Applications Developer
no	yes	hard worker	web technologies	Business Intelligence Analyst
...	...	...	...	...
yes	no	hard worker	game development	Applications Developer
yes	yes	hard worker	hacking	Business Intelligence Analyst
yes	no	smart worker	hacking	UX Designer
yes	yes	hard worker	hacking	UX Designer
yes	no	smart worker	web technologies	Software Engineer

[3780 rows x 22 columns]

Figure 10. Classified suggested job role result

Table 3. Comparison of MAE analyses using proposed L1-GCNN, K-means and Improved KSA [1] and MAEM [2] and Attn-ANN [12]

Samples	MAE			
	L1-GCNN	**K-means and Improved KSA [1]**	**MAEM [2]**	**Attn-ANN [12]**
1500	0.035	0.041	0.044	0.038
3000	0.039	0.049	0.055	0.044
4500	0.045	0.055	0.061	0.049
6000	0.048	0.058	0.064	0.053
7500	0.052	0.062	0.068	0.058
9000	0.055	0.065	0.071	0.062
10500	0.059	0.069	0.075	0.064
12000	0.063	0.073	0.079	0.068
13500	0.066	0.076	0.082	0.072
15000	0.069	0.079	0.085	0.074

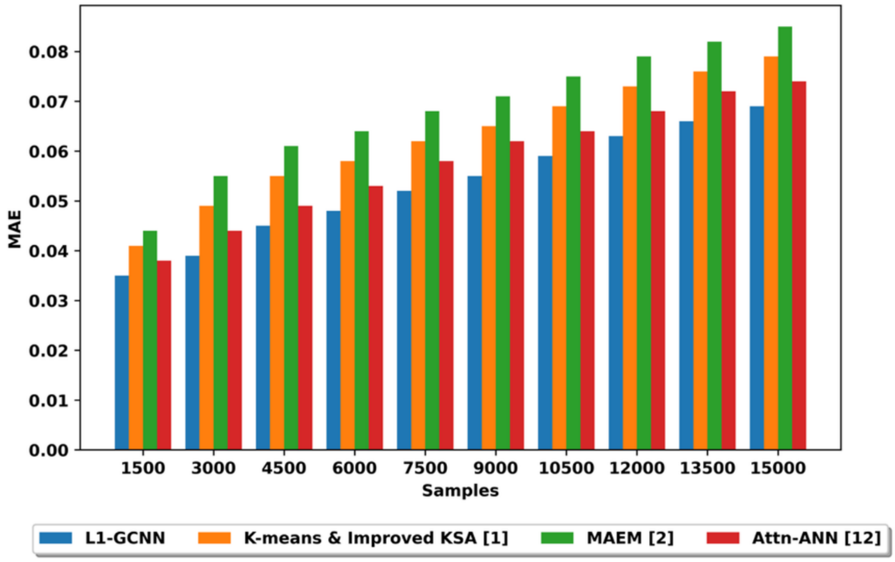

Figure 11. Graphical representation of MAE

Figure 11 shows the graphical representation of MAE using four methods: blue for the proposed L1-GCNN method, orange for K-means and Improved KSA [1], green for MAEM [2], and red for Attn-ANN [12]. A fair comparison was made using samples from the student placement dataset to evaluate four methods for MAE analysis. A steady increase in MAE was observed using all four methods. This is due to the increase in sample size; a small amount of error is said to occur, increasing the MAE rate and vice versa. However, simulations with 1500 samples yielded an MAE of 0.035 with the proposed L1-GCNN method, whereas 0.041, 0.044, and 0.038 were obtained with [1], [2], and [12], respectively. These results confirm that the MAE of the L1-GCNN method is lower than that reported in [1], [2], and [12]. The reason was that applying the normalization function and its results to Euclidean Synthetic Minority Over-sampling improves the spatial scope for generating new classes by flexibly fine-tuning the stack pointers based on local density and distribution characteristics. This, in turn, minimizes the difference between actual and predicted placement performance results for co-curricular activities using the L1-GCNN method by 19% compared to [1], 30% compared to [2], and 9% compared to [12].

Performance analysis of precision, recall, and accuracy

One of the most significant parameters used to analyze student feedback and measure placement performance is the precision rate. Precision is the ratio of relevant sample instances to the retrieved instances. This is mathematically stated as given below.

$$Pre = \frac{TP}{TP+FP} \tag{16}$$

From the above equation (16), pre 'Pre' is measured based on the true positive instances 'TP' (i.e., students with business process analyst detected as business process analyst) and the false positive instances 'FP' (i.e., students with business process analyst detected as business intelligence analyst). The recall rate is another important performance metric for analyzing student feedback to validate placement performance in co-curricular activities. Recall is the ratio of relevant instances retrieved to the total number of relevant instances. This is mathematically stated as given below.

$$Rec = \frac{TP}{TP+FN} \tag{17}$$

From the above equation (17), recall 'Rec' is measured based on the true positive rate 'TP' (i.e., students with business process analyst detected as business process analyst) and the false negative rate 'FN' (i.e., students with business intelligence analyst detected as business process analyst). Finally, accuracy is measured as given below.

$$Acc = \frac{TP+TN}{TP+TN+FP+FN} \tag{18}$$

From the above equation (18) accuracy rate 'Acc' is measured based on the true positive rate 'TP' (i.e., students with business process analyst detected as business process analyst), true negative rate 'TN' (i.e., students with business intelligence analyst detected as business intelligence analyst), false positive rate 'FP' (i.e., students with business process analyst detected as business intelligence analyst) and false negative rate 'FN' (i.e., students with business intelligence analyst detected as business process analyst) respectively. Table 4 below lists the precision, recall, and accuracy results for four methods: L1-GCNN, K-means, Improved KSA [1], and MAEM [2], as well as Attn-ANN [12].

Table 4. Comparison of Precision, recall, and accuracy analyses using proposed L1-GCNN, K-means and Improved KSA [1] and MAEM [2] and BiLSTM with Attention [12]

Samples	**Precision**				**Recall**				**Accuracy**			
	L1-GCNN	K-means and Improved KSA [1]	MAEM [2]	Attn-ANN [12]	L1-GCNN	K-means and Improved KSA [1]	MAEM [2]	Attn-ANN [12]	L1-GCNN	K-means and Improved KSA [1]	MAEM [2]	Attn-ANN [12]
1500	0.98	0.96	0.94	0.97	0.99	0.98	0.98	0.98	0.97	0.95	0.93	0.96
3000	0.94	0.81	0.70	0.9	0.98	0.85	0.79	0.94	0.94	0.80	0.77	0.9
4500	0.93	0.80	0.69	0.89	0.96	0.81	0.75	0.92	0.92	0.78	0.75	0.89
6000	0.90	0.77	0.66	0.86	0.93	0.80	0.74	0.9	0.89	0.75	0.72	0.85
7500	0.87	0.74	0.63	0.83	0.92	0.79	0.73	0.89	0.85	0.72	0.69	0.82
9000	0.85	0.72	0.61	0.81	0.90	0.77	0.71	0.87	0.83	0.69	0.66	0.8
10500	0.87	0.74	0.63	0.84	0.86	0.73	0.67	0.83	0.80	0.65	0.62	0.75
12000	0.89	0.76	0.65	0.86	0.89	0.76	0.70	0.86	0.85	0.71	0.68	0.81
13500	0.92	0.79	0.68	0.89	0.92	0.79	0.74	0.89	0.87	0.73	0.70	0.83
15000	0.94	0.81	0.70	0.9	0.94	0.76	0.70	0.91	0.89	0.75	0.72	0.86

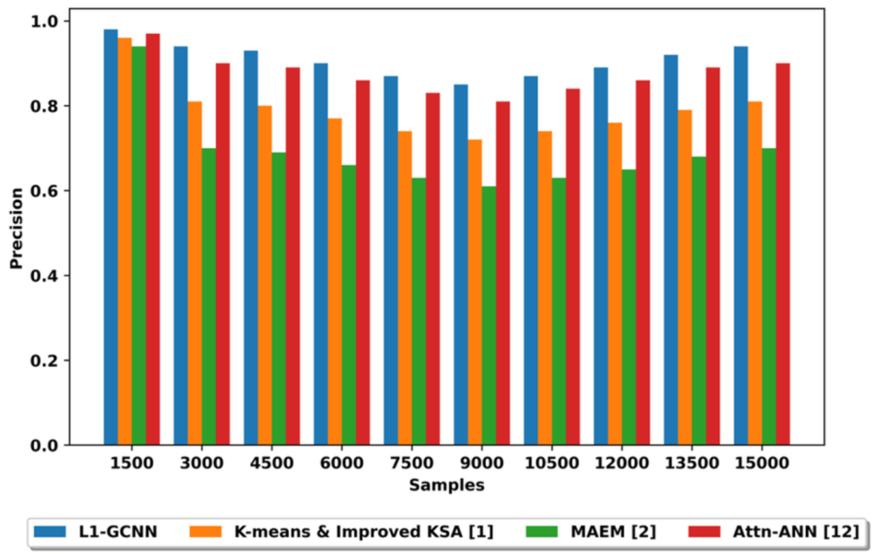

Figure 12. Graphical representation of precision

Figure 12 shows a graphical representation of precision along the vertical axis for four methods: L1-GCNN, K-means, Improved KSA [1], MAEM [2], and Attn-ANN [12]. In contrast, the samples range between 1500 and 15000 on the horizontal axis. With the blue bar representing the precision of L1-GCNN, the orange, green and blue bars denote the precision of [1] [2] and [12], respectively. From the figure above, the L1-GCNN method showed better precision than that reported in [1], [2], and [12]. The improvement in precision rate was due to applying Euclidean Synthetic Minority Over-sampling-based class balancing and then selecting the most prominent features for placement performance. With this, though traditional Synthetic Minority Over-sampling based class balancing addresses overfitting, it is found to be skewed, therefore generating biased results.

On the other hand, the proposed method, by applying Euclidean Synthetic Minority Over-sampling-based class balancing, boosts the spatial scope for generating new classes by flexibly fine-tuning their stack pointers. This is done based on local density and distribution characteristics. This generates class-balanced results that reduce the positive rate when applied to students' sentiment analysis feedback. As a result, the precision using the L1-GCNN method is 13% better than [1], 24% better than [2], and 4% better than [12].

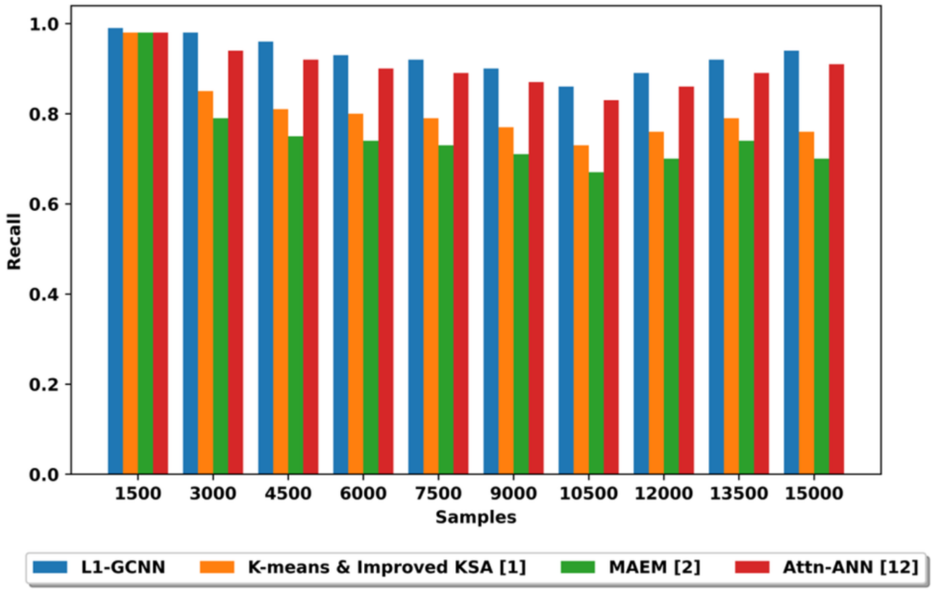

Figure 13. Graphical representation of recall

Figure 13 illustrates the graphical representation of recall on the y-axis by substituting the values in equation (17) using four different methods: L1-GCNN, K-means, Improved KSA [1], MAEM [2], and Attn-ANN [12]. On the other hand, the samples obtained from the student placement dataset are shown on the x-axis, ranging from 1500 to 15000. From the above figure, it is inferred that the recall rate obtained with the L1-GCNN method was better than those reported in [1], [2], and [12]. The improvement in recall with the L1-GCNN method was achieved by applying a graph-based convolutional neural network. Applying this graph pattern helps determine the correlation between student courses and curricular information, from which the placement performance results were derived based on supervised job roles. This association between co-curricular information and placement performance, as revealed by sentiment analysis, yields results with minimal false negatives. Therefore, the recall rate of the L1-GCNN method was improved by 16%, 25%, and 3% compared to existing methods [1] [2] and [12].

Finally, Figure 14 above shows the accuracy results when substituted using L1-GCNN, K-means, and Improved KSA [1], MAEM [2] and Attn-ANN [12]. To measure the accuracy rate, the values obtained using the three methods were substituted into equation (18), and the graphs were accordingly plotted. From the above graphical representation, the L1-GCNN method showed higher accuracy than reported in [1], [2], and [12]. The reason was that initially; class-balanced results were obtained for the supervised job role. Following this, based on the supervised job role, prominent features were selected for an L1-Regularised or logistic-regularised Graph Convolutional Neural Network model. Following this, comparison with a single layer led to the use of multiple layers to generate feature abstraction via an aggregation and transformation procedure. Finally, the most representative features for co-curricular activities were selected using the L1-Regularisation function. This, in turn, improved the overall accuracy of the L1-GCNN method by 15% compared to [1], 18% compared to [2] and 4% compared to [12], respectively.

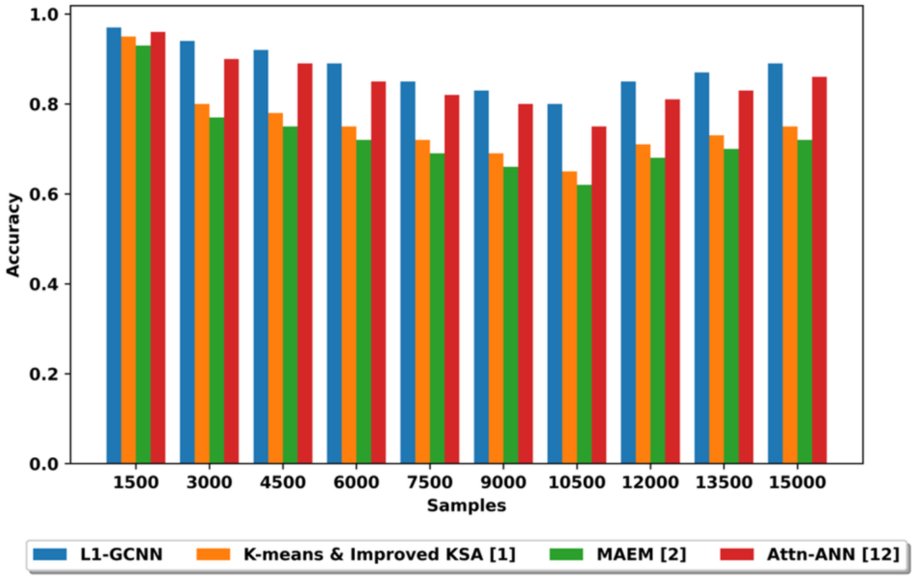

Figure 14. Graphical representation of accuracy

Comparison of proposed and existing methods using the College Student Placement Factors Dataset

The experiments are carried out for the proposed L1-GCNN method, the existing Improved KSA [1], MAEM [2] and Attn-ANN [12] using the College Student Placement Factors Dataset. Model performance is evaluated using metrics such as MAE, precision, recall, and accuracy.

Table 5. Comparison of proposed and existing methods using the College Student Placement Factors Dataset

Methods/Metrics	L1-GCNN	K-means and Improved KSA [1]	MAEM [2]	Attn-ANN [12]
Accuracy	0.921	0.892	0.834	0.904
Recall	0.936	0.905	0.811	0.913
Precision	0.895	0.864	0.812	0.887
MAE	0.062	0.074	0.079	0.069

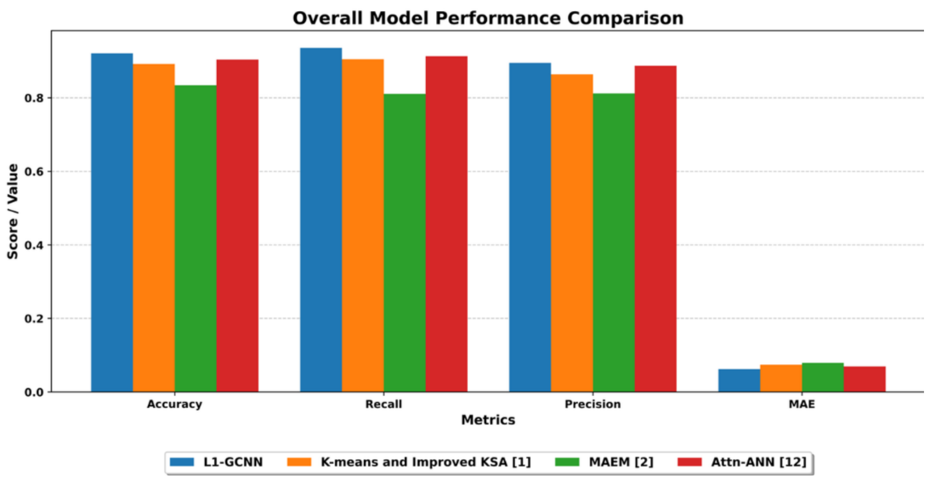

Figure 15. Results of precision, recall, accuracy and MAE for College Student Placement Factors Dataset

Table 5 and Figure 15 show the results for evaluating the proposed L1-GCNN method and the existing K-means, Improved KSA [1], MAEM [2], and Attn-ANN [12]. The PBOIDCL method achieved the highest accuracy of 0.921, precision of 0.895, recall of 0.936, and the lowest MAE of 0.062. This assessment shows that the proposed L1-GCNN method achieves high student placement performance in student sentiment analysis.

Ablation study for proposed L1-GCNN method and existing K-means, and Improved KSA [1], MAEM [2] and Attn-ANN [12] using Student Feedback Sentiment Analysis with Academic and Placement Performance dataset

The ablation study separately isolated using Pre-processing Models via L1 regularisation, GCN layers, and ESMOS. The ablation study is conducted, and the results are verified for accuracy, recall, precision, and MAE. Accuracy in the pre-processing models is measured as the percentage of data samples correctly pre-processed (see Table 6).

Table 6. Ablation study results of the L1-GCNN method and existing K-means, and Improved KSA [1], MAEM [2] and Attn-ANN [12]

Methods/Metrics	L1-GCNN	K-means and Improved KSA [1]	MAEM [2]	Attn-ANN [12]
Accuracy	0.909	0.79	0.754	0.847
Recall	0.924	0.883	0.751	0.899
Precision	0.881	0.825	0.724	0.875
MAE	0.053	0.062	0.068	0.058

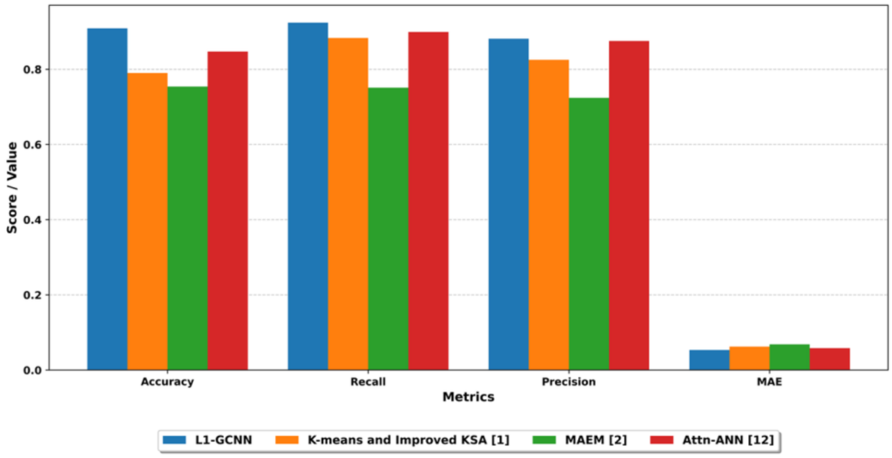

Figure 16. Graphical representation of the Ablation study results of accuracy, recall, and precision and MAE

Figure 16 shows the ablation graph for the proposed L1-GCNN method and the existing hybrid algorithm combining K-means and the Improved Krill Swarm Algorithm (KSA) (K-means and Improved KSA [1], Multi-Attribute Evaluation Model (MAEM) [2], and Attn-ANN [12]).In our work, the proposed L1-GCNN method achieved higher accuracy (0.909), precision (0.881), recall (0.924), and MAE (0.0531) than existing methods. An ablation study is conducted to examine the significance of each contribution involved in the proposed methods. The pre-processed data are given as input to the feature selection process. In this step, the spatial and temporal features are selected. Using the selected features, classification is performed to accurately classify the samples.

One-way ANOVA test

One-way ANOVA (Analysis of Variance) is a statistical test used to determine whether there is a significant difference among two or more groups in a dataset. Tukey's Honestly Significant Difference (HSD) test is a post hoc analysis employed. One-way ANOVA test is measured for the proposed L1-GCNN method and existing K-means and Improved KSA [1], MAEM [2] and Attn-ANN [12] with metrics (i.e., accuracy) by using Student Feedback Sentiment Analysis with Academic and Placement Performance dataset and College Student Placement Factors Dataset.

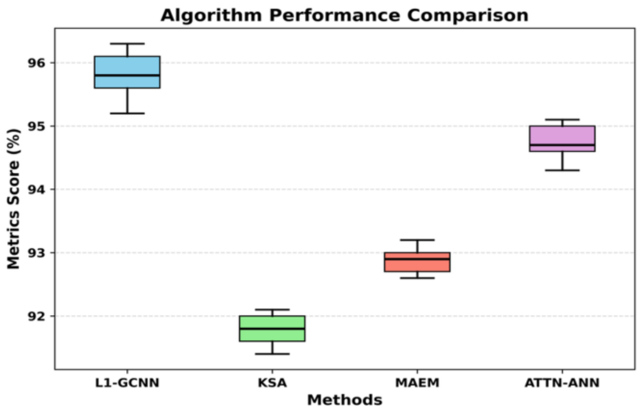

Figure 17. One-way ANOVA statistical test results for Student Feedback Sentiment Analysis with Academic and Placement Performance dataset

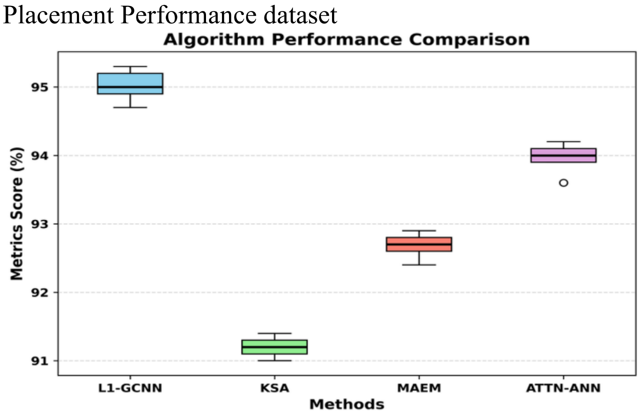

Figure 18. One-way ANOVA statistical test results for College Student Placement Factors Dataset

Figures 17 and 18 present the results of the One-way ANOVA. The x-axis shows methods such as the proposed L1-GCNN and existing K-means, Improved KSA [1], MAEM [2], and Attn-ANN [12], and the y-axis shows the metric score (accuracy). From the results, the F-statistic is 153.16, and the p-value is 0.05, using Student Feedback Sentiment Analysis with the Academic and Placement Performance dataset. From the results, the F-statistic is 316.25, and the p-value is 0.02 using the College Student Placement Factors Dataset. The proposed L1-GCNN model achieved better performance than other existing methods.

5. CONCLUSIONS

Campus placements in academic or co-curricular activities are pivotal in today's competitive job market, helping students transition from academia to professional careers. Nevertheless, conventional placement procedures for co-curricular activities pose challenges due to a lack of data-driven insights, leaving students uncertain about their employability. This paper proposes the L1-regularised Graph Convolutional Neural Network (L1-GCNN) for measuring placement performance based on student feedback from co-curricular activities. L1-GCNN adopts Euclidean Synthetic Minority Over-sampling-based Pre-processing, thereby reducing MAE. Encapsulating interdependencies between co-curricular metrics and student placement performance via a feedback graph structure based on L1-regularised or logistic function is applied, improving sentiment analysis precision and recall. L1-GCNN then efficiently leverages a graph convolutional neural network to accurately capture placement performance results. Experiments on the student placement dataset show that the L1-GCNN method outperforms all baselines for MAE, precision, recall, and accuracy.

The limitation of Student Sentiment Analysis with Co-Curricular Activities and Placement is that sentiment models struggle to understand sarcasm, irony, or culturally specific expressions, leading to erroneous sentiment classification. The proposed model trained on biased datasets may not accurately reflect diverse student experiences, and a general lexicon may fail to capture domain-specific language related to co-curricular activities or placements. The educators are offering data-driven feedback to improve programs and employers by identifying and engaging with qualified talent.

REFERENCES

[1] Z. Dai, "Analysis of college students' physical education learning effect and quality based on text data fusion," Adv. Continuous Discrete Models, vol. 25, no. 64, pp. 1–15, Mar. 2025, doi: 10.1186/s13662-025-03925-9.

[2] H. Liang and H. Liang, "Deep learning based multi-attribute evaluation for holistic student assessment in physical education," Sci. Rep., vol. 15, no. 1, pp. 1–17, May 2025, doi: 10.1038/s41598-025-02168-8.

[3] A. L. Hemdanou, M. L. Sefian, Y. Achtoun, and I. Tahiri, "Comparative analysis of feature selection and extraction methods for student performance prediction across different machine learning models," Comput. Educ. Artif. Intell., vol. 7, Art. no. 100301, Dec. 2024, doi: 10.1016/j.caeai.2024.100301.

[4] V. L. Williams, M. Haddad, K. A. Kelley, J. Li, and J. L. Rodis, "Evaluation of cocurricular learning and the influence of student choice," Am. J. Pharm. Educ., vol. 88, no. 8, Art. no. 101257, Sep. 2024, doi: 10.1016/j.ajpe.2024.101257.

[5] C. S. Moreno Cortez and J. Nevarez Montes, "Extracurricular arts: Effects on creativity and academics," J. Creativity, vol. 35, no. 1, Art. no. 100094, Apr. 2025, doi: 10.1016/j.yjoc.2025.100094.

[6] W. Strielkowski, V. Grebennikova, A. Lisovskiy, G. Rakhimova, and T. Vasileva, "AI-driven adaptive learning for sustainable educational transformation," Sustain. Dev., vol. 32, no. 2, pp. 1921–1947, Oct. 2024, doi: 10.1002/sd.3221.

[7] X. Li, "Student achievement prediction and auxiliary improvement method based on fuzzy decision support system," Discov. Artif. Intell., vol. 5, pp. 1–25, Apr. 2025, doi: 10.1007/s44163-025-00308-7.

[8] A. Olewnik, Y. Chang, and M. Su, "Co-curricular engagement among engineering undergrads: Do they have the time and motivation?" Int. J. STEM Educ., vol. 10, no. 1, pp. 1–20, Oct. 2023, doi: 10.1186/s40594-023-00410-1.

[9] E. Pürgstaller, J. Schmeling, L. Barkemeyer, M. Ullmann, and K. Koal, "Attitudes of physical education teachers and student teachers regarding digital media in physical education," Ger. J. Exerc. Sport Res., vol. 55, pp. 551–554, Apr. 2025, doi: 10.1007/s12662-025-01035-y.

[10] C. Weng, C. Chen, and X. Ai, "A pedagogical study on promoting students' deep learning through design-based learning," Int. J. Technol. Des. Educ., vol. 33, pp. 1653–1674, Nov. 2022, doi: 10.1007/s10798-022-09789-4.

[11] J. Yu and C. Qi, "Machine learning-based sentiment analysis in English literature: Using deep learning models to analyse emotional and thematic content in texts," IEEE Access, vol. 13, pp. 1–12, Apr. 2025, doi: 10.1109/ACCESS.2025.3553386.

[12] S. Leelaluk, T. Minematsu, C. Tang, Y. Taniguchi, F. Okubo, A. Shimada, and T. Yamashita, "Attention-based artificial neural network for student performance prediction based on learning activities," IEEE Access, vol. 12, pp. 100659–100675, Jul. 2024, doi: 10.1109/ACCESS.2024.3429554.

[13] Rahman, S. R., Islam, M. A., Akash, P. P., Parvin, M., Moon, N. N., & Nur, F. N., "Effects of co-curricular activities on students' academic performance by machine learning," Curr. Res. Behav. Sci., vol. 2, Art. no. 100057, Nov. 2021, doi: 10.1016/j.crbeha.2021.100057.

[14] D. Ileritürk, "Evaluation of extracurricular activities in education according to pre-school teacher candidates' views," Soc. Sci. Humanit. Open, vol. 8, no. 1, Art. no. 100524, Oct. 2023, doi: 10.1016/j.ssaho.2023.100524.

[15] H. V. Le, "Factors impeding university students' participation in English extracurricular activities: Time constraints and personal obstacles," Heliyon, vol. 10, no. 5, Art. no. e27332, Mar. 2024, doi: 10.1016/j.heliyon.2024.e27332.

[16] A. Milicevica, M. Despotovic-Zrakic, D. Stojanovic, M. Suvajzic, and A. Labus, "Academic performance indicators for the hackathon learning approach—The case of the blockchain hackathon," J. Innov. Knowl., vol. 9, no. 3, Art. no. 100501, Jun. 2024, doi: 10.1016/j.jik.2024.100501.

[17] E. Kalita, A. M. Alfarwan, H. El Aouifi, A. Kukkar, S. Hussain, T. Ali, and S. Gaftandzhieva, "Predicting student academic performance using Bi-LSTM: A deep learning framework with SHAP-based interpretability and statistical validation," Front. Educ., vol. 10, pp. 1–18, 2025, doi: 10.3389/feduc.2025.1581247.

[18] S. G. K. Salee, M. Asif, F.-E. Amin, and M. Imran, "Advancing aspect-based sentiment analysis in course evaluation: A multi-task learning framework with selective paraphrasing," IEEE

Access, vol. 13, pp. 7764–7779, Jan. 2025, doi: 10.1109/ACCESS.2025.3527367.
[19] J. Pan, Z. Zhao, and D. Han, "Academic performance prediction using machine learning approaches: A survey," IEEE Trans. Learn. Technol., vol. 18, pp. 351–368, Jan. 2025, doi: 10.1109/TLT.2025.3554174.
[20] Y. Mamani-Coaquira and E. Villanueva, "A review on text sentiment analysis with machine learning and deep learning techniques," IEEE Access, vol. 12, pp. 193115–193130, Dec. 2024, doi: 10.1109/ACCESS.2024.3513321.
[21] I. S. Hueck, A. Guével, R. S. MacLeod, and K. Billiar, "Integration of co-curricular experiential learning in BME programs to increase student success," Biomed. Eng. Educ., vol. 5, pp. 189–202, Apr. 2025, doi: 10.1007/s43683-025-00183-9.
[22] R. Singh, S. K. Singh, and N. Mishra, "Factors influencing student learning performance and continuous use of artificial intelligence in online higher education," Discov. Educ., vol. 4, no. 292, pp. 1–18, Jun. 2025, doi: 10.1007/s44217-025-00728-8.
[23] D. Ç. Ertuğrul and S. Bitirim, "Job recommender systems: A systematic literature review, applications, open issues, and challenges," J. Big Data, vol. 12, no. 1, pp. 1–111, Jul. 2025, doi: 10.1186/s40537-025-01173-y.
[24] C. C. Thelma, E. V. Phiri, M. Morgan, and M. M. Gilbert, "The effect of co-curricular activities on learners' academic performance: A case of selected secondary schools in Lusaka District, Zambia," Int. J. Res., vol. 11, no. 8, pp. 243–265, Aug. 2024, doi: 10.5281/zenodo.13335567.
[25] A. W. O'Donnell, G. Redmond, A. A. Gardner, J. J. J. Wang, and A. Mooney, "Extracurricular activity participation, school belonging, and depressed mood: A test of the compensation hypothesis during adolescence," Appl. Dev. Sci., vol. 28, pp. 596–611, 2024, doi: 10.1080/10888691.2023.2260745.
[26] F. Lin, C. Li, R. W. Y. Lim, and Y. H. Lee, "Empower instructors with actionable insights: Mine and visualize student written feedback for instructors' reflection," Comput. Educ. Artif. Intell., vol. 8, Art. no. 100389, Jun. 2025, doi: 10.1016/j.caeai.2025.100389.
[27] S. Hooshangi, R. Buxton, and M. Ellis, "Integration of practical computing skills and co-curricular activities in the curriculum," in Proc. ACM Conf., Jul. 2022, pp. 61–67, doi: 10.1145/3502718.3524802.
[28] N. Wang and Q. Wang, "Modality-uncertainty-aware knowledge distillation framework for multimodal sentiment analysis," Complex Intell. Syst., vol. 12, 2026, doi: 10.1007/s40747-025-02135-w.
[29] W. Saidi, B. Satouri, and A. El Abderrahmani, "Optimal architecture for a sentiment analysis transformer with multihead attention and genetic crossover," Sci. Rep., vol. 15, Art. no. 37554, 2025, doi: 10.1038/s41598-025-22064-5.
[30] C. Yu and X. Yu, "Application of mobile learning system based on convolutional network technology in students' open teaching strategies," Sci. Rep., vol. 15, Art. no. 41646, 2025, doi: 10.1038/s41598-025-25532-0.
[31] K. S. Prasad, "Student placement data," Kaggle, 2020. [Online]. Available:https://www.kaggle.com/datasets/koshikasaiprasad/student-placement-data/data.
[32] S. Islam, "College student placement factors dataset," Kaggle, 2025. [Online]. Available:https://www.kaggle.com/datasets/sahilislam007/college-student-placement-factors-dataset.